

Performance Analysis of Flagged BCD Adder and Pipelined BCD Adder

Ullas. S. S, S.S. Ravishankar

Department of Electronics and Communication, S.D.M College of Engineering and Technology, Dharwad, Karnataka, India

ABSTRACT

BCD Adder is the fundamental adder we learn in logic design and any basic electronics lab. Conventional BCD Adder what we generally know is not feasible for higher bits as they require more area and as they have more propagation delay for higher bit extension. For higher level application BCD design we require more efficient basic BCD block. So in this paper we proposed 2 types of BCD adders namely flagged BCD adder and pipelined BCD adder which will overcome the disadvantages of previous design and in this we will analysis the performance of all these BCD adders in terms of speed, power, area using Xilinx 14.5 and we used Verilog language for coding purpose.

Keywords: BCD adders, Parallel Adders, Pipelining Adder, Computer Arithmetic, Power, Delay

I. INTRODUCTION

Binary addition is a standout amongst the most primitive and most usually utilized applications as a part of PC math .With the fast development of decimal math in numerous applications for example, business, monetary, or web field. The utilization of least complex and simple strategy for decimal number-adding gets to be important for the planners and clients. Previously decades, despite the fact that the binary number adding is broadly utilized as a part of processors or whatever other applications, yet some issue happened in performing some binary math operations. To start with is the portion numbers can't be spoken to by utilizing the binary numbers. It will require endless bits for representation, so it get to be erroneous decimal divisions. This erroneous representation of decimal division causes guess mistakes.

Furthermore, second is ,Financial database contain decimal information in the event that we are utilizing paired equipment then first decimal information is changed over into paired and after calculation result which is in parallel from it again change over in decimal information these transformation will build the spread deferral. In this way, to conquer this downside of parallel number juggling, the Binary Coded Decimal numbers is utilized. In BCD, every piece of decimal

numbers 0 to 9 utilizes four bits 00002 to 10012. BCD operations can be productive when perusing from a BCD gadget, doing a straightforward math operation (e.g., a solitary expansion) and afterward composing the BCD worth to some other gadget. Numerous designs and calculations have been proposed to date for decimal math.

To further decrease power and area in BCD addition a new BCD adder is proposed utilizing hailed double expansion for the rectification consistent expansion .The yield of adders of first arrange and hailed calculation piece are gone through a multiplexer. The control signal for the multiplexer is created from a control circuit which delivers 1 for entirety values surpassing 9 and 0 else. In any case, because of the utilization of the multiplexer the engendering delay is expanded [13]. To diminish the restriction of this BCD adder we proposed a new BCD adder which speed is quick then these BCD adders .This paper is sorted out as takes after. In area II, the typical BCD adder, in segment III hailed rationale BCD adder are quickly inspected. Area IV portrays the proposed high speed BCD adders. Segment V depicts the proposed 64 bit pipelined BCD adder .segment VI depicts the execution results and the point by point correlation of all sorts BCD adders. At last, area VII closes this paper

II. METHODS AND MATERIAL

Basic BCD ADDER

In a BCD adder assume we have two information X and Y are given to BCD adder engineering is appeared in figure 1. After utilizing these first set adders which created by 4 back to back full adders to include the estimations of info X and Y, the digit adder with adjustment which is likewise made by 4 full adders is utilized. At the point when consequence of entirety is more than 9 then we include (0110)₂ in every snack by utilizing adjustment system. Remedy values 0110, is dictated by the yield of $c + (S[3] \cdot S[2]) + (S[2] \cdot S[1])$, But the BCD adder is exceptionally straightforward, additionally ease back because of the convey progressively outstretching influence. It likewise utilized two paired adders first to include information and second is used to include correction in the yield of first paired adder because of this reason it builds engendering delay and area.

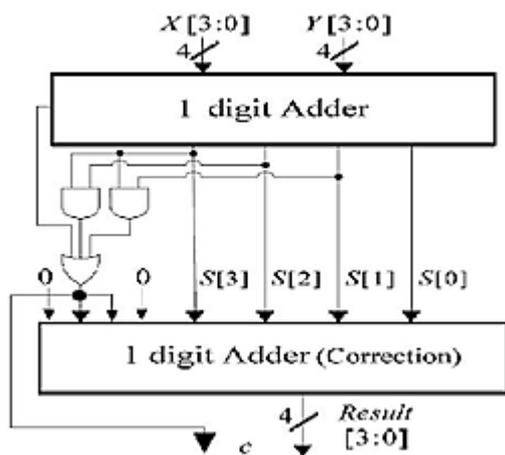


Figure 1: Normal BCD Adder

FLAGGED BCD ADDER

To diminish the limitation of normal BCD adder another BCD adder was designed. The various parts of the proposed BCD adder are 4bit Ripple Carry Adder(RCA), Abundance 9 detector, Flag bit Calculating block, signal inversion piece and four 2:1 multiplexers whose schematic is shown in figure 2. The information an (a3a2a1a0) and B (b3b2b1b0) are encouraged to the excess 9 detector. The entirety output S (S3S2S1S0) and do Co of this stage is bolstered to Excess 9 detector. On

the off chance that the total S(S3S2S1S0) is not exactly or equal to 9 the Cout of Excess 9 locator will be zero and the sum S(S3S2S1S0) will be passed out through the Multiplexer. If the sum S (S3S2S1S0) surpasses 9, the Cout of Excess 9 detector will be 1 and the sum bits will be passed through the flag bit computation block to create entomb mediate carry bits (d4d3d2d1)

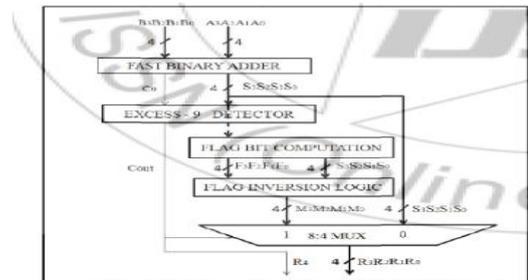


Figure 2: FLAGGED BCD ADDER

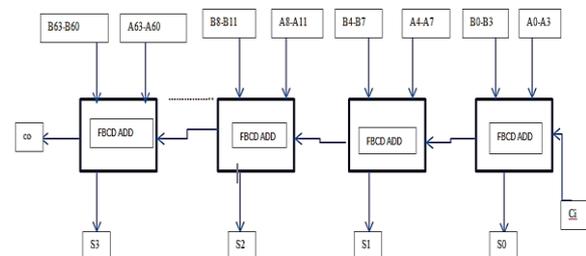


Figure 3: 64 Bit Structural Calling of Flagged BCD Adder

The convey b it (d4d3d2d1) and entirety S(S 3S2S1S0) are then utilized by this square to quality ate banner bits (F0,F1,F2,F3) The banner bits(F0,F1,F2,F3) and aggregate S(S3S2S1S0) are passed through banner reversal rationale to generate the BCD output M3 M2 M1 M0 for S (CoS3S2 S1S0) which exceeds 9. The M3 M2 M1 M0 of the hailed reversal alliance k shapes the other info to the multiplexer which is passed out for 1 value of Cout. The hailed BCD adder beat a ll different previous designs as far as postponement and zone. This same 4 bit structure is used to design 64 bit Design using structural description.

The Flagged BCD adder also has some restriction like it utilized a multiplexer as a part of definite stage which expanded the spread way so propagation delay is additionally increased.

If we refer the previous block diagram then there will be a block called Fast Binary Adder. We know that Ripple

Carry Adder although very simple to design but it has more delay due to carry ripple effect so if we use RCA then there will be combined effect of mux and RCA and overall delay will become more so instead of RCA we can also use higher adders like RCA,RLA and Parallel Adders.

In this adder we have used RCA,RLA and parallel adders like KS ,BK,HC,LF adders and results are compared.

PIPELINED BCD ADDER

Although Flagged BCD Adder overtakes normal BCD adder in terms of area and power but it has more propagation delay. So in order to overcome this disadvantage new adder is designed where we use flagged BCD adder as a 4 bit fast binary adder. Here we all know that it also consist two main stages one is adding two inputs and second one is generating the flagged bits instead of correction bits.

While designing this type of adder pipelining is introduced between each block of full adder in first stage. Pipelining is defined as the implementation technique where multiple instructions are overlapped in execution. There are many types of Pipelining strategies, but in this paper we use simple strategy of introducing a delay flip-flops or registers between intermediate stages This will reduce the propagation delay due to long structural calling and after introducing the registers each stage will become independent by working on three values on same time and propagation delay of this type becomes smaller than the previous 64 bit design using 4 bit Flagged BCD Adder.

The next diagram shows the 4bit Flagged BCD adder and the pipelined BCD adder structure.

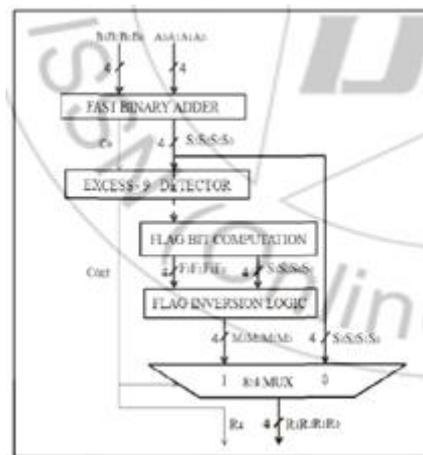


Figure 4: 4 bit Flagged BCD Adder

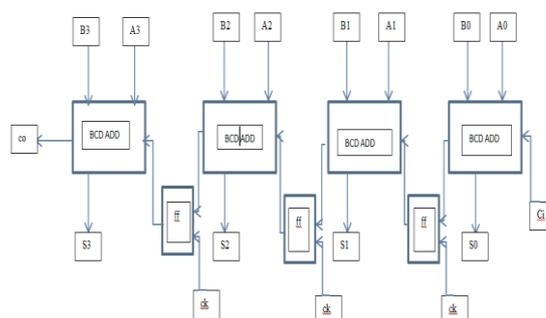


Figure 5: Proposed 4 bit Pipelined Structure

In the above fig4 the ‘ff’ denotes the flip flop and ck denotes the clock input.

III. RESULTS AND DISCUSSION

In this part we will discuss about the results of different types of BCD adders and their performances. In this paper we used Xilinx 14.5 as a tool on Spartan 6 platform, so all the results correspond to Spartan 6 with speed -4 components. For the coding purpose we have used Verilog as Hardware Description Language. In this paper we have found the area in terms of LUT’s , Speed in terms of propagation delay measured in nano seconds and power in terms of milliwat (mW)

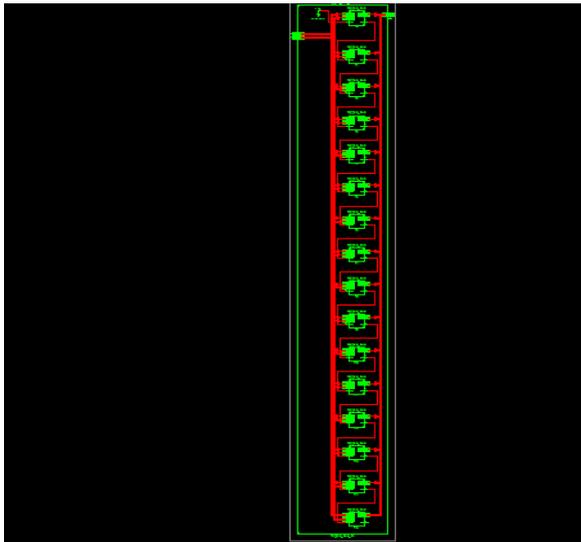
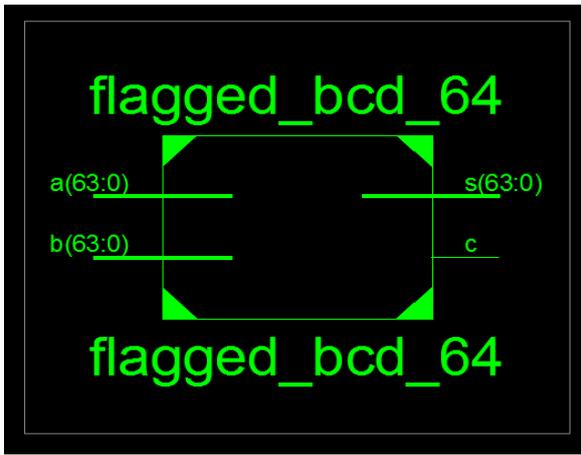


Figure 6: RTL Schematics of Flagged BCD Adder

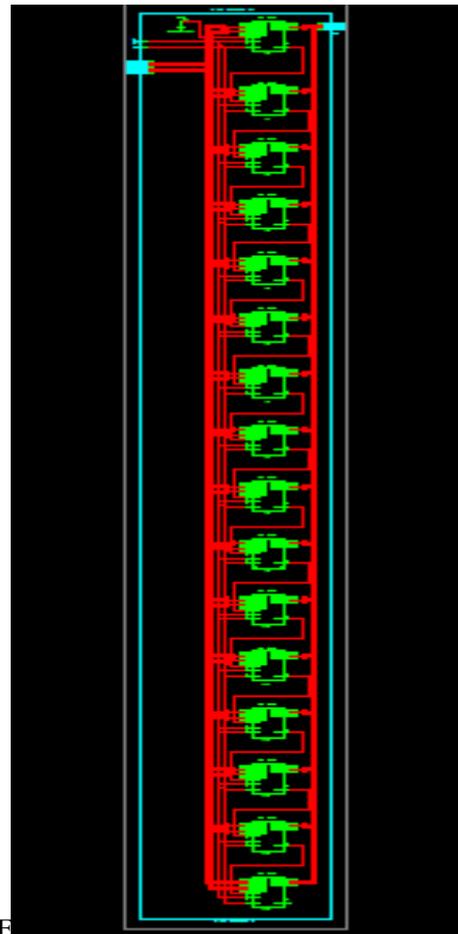
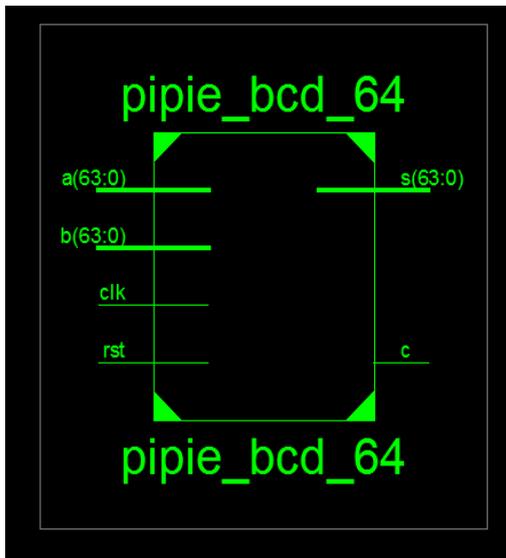


Figure 7: RTL Schematics of Pipelined BCD Adder

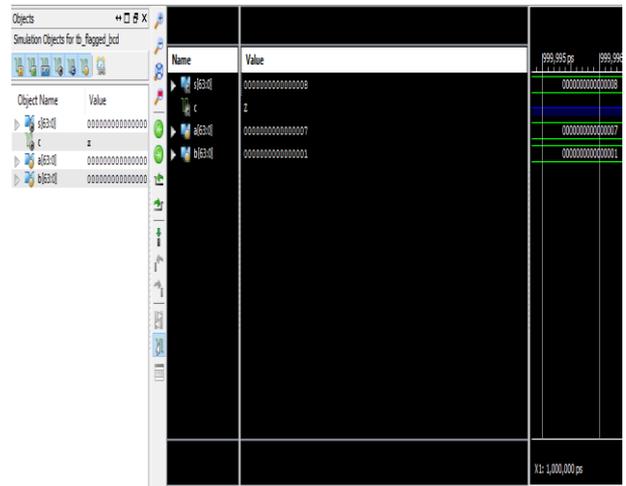


Figure 8: Simulation Result of Flagged BCD Adder

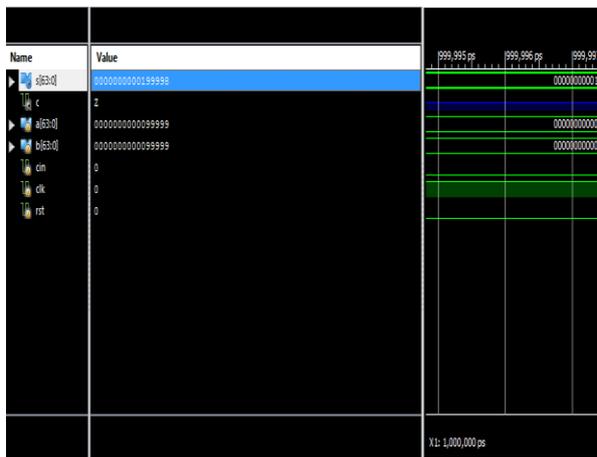


Figure 9: Simulation Result Of Pipelined BCD Adder

Power, Area, Speed Summary

	Power(mW)	Area(LUT's)	Speed(ns)
Flagged rc	21	126	39.974
Flagged ks	21	127	7.521
Flagged bk	21	144	8.727
Flagged lf	21	208	8.962
Flagged hc	21	208	8.962
Pipelined BCD	44	190	7.229

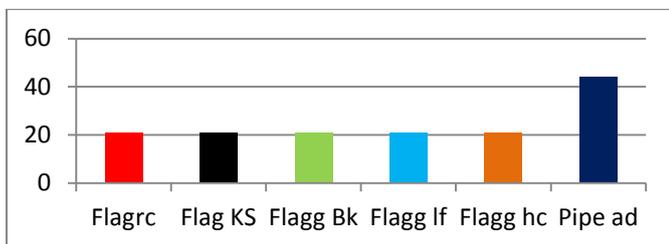


Figure 10 : Power Comparison of Various Adders

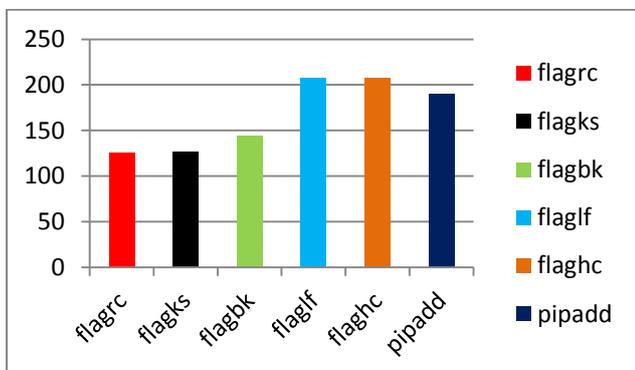


Figure 11: Area Comparison of various adders

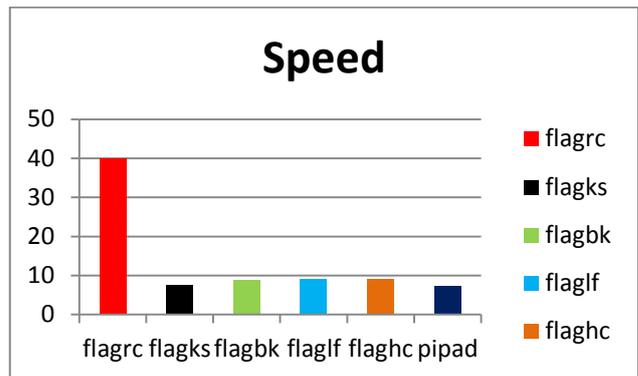


Figure 12: Speed Comparison of various adders

IV. CONCLUSION

In this paper we have seen mainly 2 types BCD Adder which will overcome the disadvantages of the conventional BCD adder. The 3 constraints of the VLSI design is speed, power and area so we have examined all 3 constraints for our design using Spartan 6 with speed grade -3 and all the results are summarized in table and comparison is done using stock chart. We also 4 parallel prefix adders for flagged bcd adder to reduce the propagation delay.

V. REFERENCES

- [1] Tso-Bing Juang, Hsin-Hao Peng, Chao-Tsung Kuo. "Area efficient BCD Adder," 19th IEEE/IFIP International Conference on VLSI Design, 2011 Draft IEEE Standard for Floating-Point Arithmetic. New York: IEEE, Inc., 2004.
- [2] Alp Arslan Bayrakci and Ahmet Akkas. Reduced Delay BCD Adder. IEEE, 2007. T. Lang, and A. Nannarelli, "Division Unit for Binary Integer Decimals," Prof. 20th IEEE International Conference on Application-specific Systems, Architectures and Processors (ASAP), pp. 1-7, 2009.
- [3] T. Lang and A. Nannarelli, "A Radix-10 Digit-Recurrence Division Unit: Algorithm and Architecture," IEEE Transactions on Computers, Vol. 56, No. 6, pp. 727-739, June 2007.
- [4] R. K. James, T. K. Shahana, K. P. Jacob, and S. Sasi, "Decimal multiplication using compact BCD multiplier," Proc. International Conference on Electronic Design (ICED), pp. 1-6, 2008
- [5] P. M. Kogge and H. S. Stone. A Parallel Algorithm for The Efficient Solution of a General Class of Recurrence Equations. IEEE Trans. on Computers, C-22(8), Aug. 1973.
- [6] M. M. Mano. Digital Design, pages 129–131. Prentice Hall, third edition, 2002.
- [7] M. S. Schmookler and A.W. Weinberger. High Speed Decimal Addition. IEEE Transactions on Computers, C-20:862– 867, Aug. 1971.
- [8] B. Shirazi, D. Y. Y. Young, and C. N. Zhang. RBCD: Redundant Binary Coded Decimal Adder. In IEEE Proceedings, Part E, No. 2, volume 136, pages 156–160, March 1989.
- [9] J. D. Thompson, N. Karra, and M. J. Schulte. A 64-Bit Decimal Floating-Point Adder. In Proceedings of the IEEE Computer Society Annual Symposium on VLSI, pages 297–298, February 2004.
- [10] +B. Prashanthi kumari, G. N. V. Ratna Kishor, "New Approach for Implementing BCD Adder Using Flagged Logic" International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 12, December - 2013